

1 Objectivos - Uso e implementação de Listas Duplamente Ligadas

Pretende-se que apresente uma implementação do tipo de dados (*BigInt*), que representa inteiros que ultrapassem a ordem de grandeza dos inteiros do C (*Integer* e *Long*). Deverá ser possível realizar as operações básicas de adição, subtracção, multiplicação, divisão e resto. A implementação deste tipo de inteiros deve ser realizada à custa de listas ligadas.

2 O Trabalho

O trabalho consiste na apresentação duma biblioteca constituída por 5 ficheiros: dois ficheiros *.h* 3 ficheiros *.c* : Ficheiros *.h*:

- *bigInt.h*, ficheiro com as definições do tipo *BigInt*, criação de *BigInts*, e as operações sobre *BigInts* que devem ser implementadas
- *doubleList.h* ficheiro com as definições do tipo *DList* correspondente a listas duplamente ligadas

Para que não hajam dúvidas sobre as definições de ambos os tipos, são apresentados excertos dos ficheiros *.h*. Na sua implementação de *BigInt* deve providenciar a criação dum *BigInt*, através duma string de dígitos. Por exemplo a string "5675300255355316716367366371" dará origem ao *BigInt* 5675300255355316716367366371.

```
8  #ifndef bigInt_h
9  #define bigInt_h
10
11  #include <stdio.h>
12
13
14
15
16  BigInt big_new(char *num);
17  BigInt sum_b(BigInt a, BigInt b);
18  BigInt sub_b(BigInt a, BigInt b);
19  BigInt mult_b(BigInt a, BigInt b);
20  BigInt div_b(BigInt a, BigInt b);
21  BigInt mod_b(BigInt a, BigInt b);
22  void print_b(BigInt a);
23
24
```

Figure 1: Excerto do ficheiro *bigInt.h*

```

struct DNode;
struct DListStruct;
typedef struct DNode *DPosition;
typedef struct DListStruct *DList;

DList CreateDList( void );
void MakeEmptyDList( DList L );
int SizeDList(DList L);
DPosition DHeader(DList L);
DPosition DFooter(DList L);
int IsEmptyDList(DList L);
void InsertDList(ElementType X, DPosition P);
void InsertDListIth(ElementType X, int i, DList L);
void addDList(ElementType X, DList L);
DPosition FindDList(ElementType e);
void DeleteElement(ElementType e, DList L);
ElementType RemoveElementAt(int i, DList L);
DPosition Advance(DPosition P);
DPosition Back(DPosition P);
ElementType Retrieve(DPosition P);
void PrintDList(char *name, DList L);

#endif /* doubleLList_h */

```

Figure 2: Excerto do ficheiro *doubleLList.h*

O último ficheiro a apresentar será um ficheiro que permita criar e manipular *BigInts*. A título de exemplo:

```

int main(int argc, const char * argv[]) {
    // insert code here...
    printf("Hello, World!\n");
    BigInt a=big_new("-1453778288677");
    BigInt b=big_new("898385199936628789819168712");
    printf("a=");
    print_b(a);
    printf("b=");
    print_b(b);
    BigInt soma=sum_b(a,b);
    printf("soma=");
    print_b(soma);
    return 0;
}

```

Figure 3: Excerto do ficheiro *main.c*

Note que no exemplo apresentado nem sequer aparecem as listas. De qualquer modo, deverá ser possível manipular directamente as listas por inclusão no ficheiro main.c do header correspondente.

2.1 Entrega

O trabalho será realizado individualmente. A data limite para a submissão do trabalho é dia 3 de Maio de 2022, sendo realizada a submissão pelo moodle, nos moldes habituais. Todos os ficheiros deverão ser "zipados" e submetidos num único ficheiro com o número do aluno que realizou o trabalho e o nome do trabalho(ex: "NNNNN-big.zip").